

IN THE CLAIMS:

Please substitute the following claims for the same-numbered claims in the application:

1. (Currently Amended) A method of data replication in a distributed computing system, said method comprising:

assigning a delta production/consumption value for arbitrary data sources and targets operable for replicating data;

embedding replication tracking information within said data, wherein said replication tracking information comprises a timestamp and a contiguous sequence number; and

atomically and independently applying updates at a target site using said replication tracking information[[.]]; and

using an apply service at said target site to embed and analyze said tracking information during a crash recovery sequence, wherein said apply service utilizes an in-memory index when a system crash occurs and a recovery process is initiated by said distributed computing system.

2. (Original) The method of claim 1, further comprising using a capture service at a source site for flow control, wherein said capture service comprises a buffer.

3. (Currently Amended) The method of claim [[1]] 2, ~~further comprising using an apply service at said target site to embed and analyze said tracking information during a crash recovery sequence~~ wherein said capture service is adapted to prevent resending of an already sent delta

production/consumption value from a source site to said target site.

4. (Original) The method of claim 1, further comprising using a monitor service to maintain a state of ongoing replications for status and quality-of-service tracking.
5. (Original) The method of claim 1, further comprising allowing data sources and targets of arbitrary data formats, including relational DBMSs, files, query results, XML DBMSs to be replicated, through an abstraction of delta (change) production/consumption, and a monotonically increasing timestamp on each said delta.
6. (Original) The method of claim 1, wherein said replication tracking information is used to determine if a given delta has been previously applied to said target site.
7. (Original) The method of claim 1, wherein in an event of a crash in said system, said target site requests retransmission of replicated data from said source site beginning at a given timestamp and sequence number.
8. (Original) The method of claim 1, wherein said sequence number and timestamp are operable to determine if any transaction has been lost during transmission from said source site to said target site, wherein said sequence number is a contiguous series of numbers increasing from 1 to n and said timestamp is any monotonically increasing sequence of numbers.

9. (Original) The method of claim 1, wherein said target site is operable to apply deltas autonomously and independently from said source site.

10. (Currently Amended) The method of claim [[1]] 2, wherein said capture and apply services send periodic signals to said monitor service to track a progression of replication for answering status and quality of service queries.

11. (Currently Amended) The method of claim [[1]] 2, wherein said capture service selectively removes replication requests which lag other requests by more than a predetermined permissible amount.

12. (Original) The method of claim 1, wherein said replicated data further comprises origination tags, wherein said origination tags are operable to prevent duplicate replications of a same data from occurring at said target site via different routes.

13. (Original) The method of claim 1, wherein said apply service utilizes run-length encoding to compactly describe an interval of timestamps and sequence numbers.

14. (Currently Amended) The method of claim [[1]] 2, ~~wherein said apply service utilizes an in-memory index when a system crash occurs and a recovery process is initiated by said distributed computing system~~ wherein communication between said capture service and said apply service occurs over an unreliable user datagram protocol (UDP) communication channel.

15. (Original) The method of claim 1, wherein said target site autonomously tracks a progression of replication of said data by maintaining a separate table of applied deltas.

16. (Original) The method of claim 15, wherein said separate table comprises an entry, wherein each entry in said table comprises said timestamp and said sequence number of a delta, and wherein said sequence number is operable to determine if a transaction has been misplaced in said distributed computing system.

17. (Original) The method of claim 15, wherein a file-based target site can maintain said table in a separate file and perform atomic updates by writing said file to a disk before updated files are written to said disk.

18. (Currently Amended) A program storage device readable by computer, tangibly embodying a program of instructions executable by said computer to perform a method of data replication in a distributed computing system, said method comprising:

assigning a delta production/consumption value for arbitrary data sources and targets operable for replicating data;

embedding replication tracking information within said data, wherein said replication tracking information comprises a timestamp and a contiguous sequence number; ~~and~~

atomically and independently applying updates at a target site using said replication tracking information[[]]; and

using an apply service at said target site to embed and analyze said tracking information during a crash recovery sequence, wherein said apply service utilizes an in-memory index when a system crash occurs and a recovery process is initiated by said distributed computing system.

19. (Original) The program storage device of claim 18, further comprising using a capture service at a source site for flow control, wherein said capture service comprises a buffer.

20. (Currently Amended) The program storage device of claim ~~[[18]]~~ 19, ~~further comprising using an apply service at said target site to embed and analyze said tracking information during a crash recovery sequence~~ wherein said capture service is adapted to prevent resending of an already sent delta production/consumption value from a source site to said target site.

21. (Original) The program storage device of claim 18, further comprising using a monitor service to maintain a state of ongoing replications for status and quality-of-service tracking.

22. (Original) The program storage device of claim 18, further comprising allowing data sources and targets of arbitrary data formats, including relational DBMSs, files, query results, XML DBMSs to be replicated, through an abstraction of delta (change) production/consumption, and a monotonically increasing timestamp on each said delta.

23. (Original) The program storage device of claim 18, wherein said replication tracking information is used to determine if a given delta has been previously applied to said target site.

24. (Original) The program storage device of claim 18, wherein in an event of a crash in said system, said target site requests retransmission of replicated data from said source site beginning at a given timestamp and sequence number.

25. (Original) The program storage device of claim 18, wherein said sequence number and timestamp are operable to determine if any transaction has been lost during transmission from said source site to said target site, wherein said sequence number is a contiguous series of numbers increasing from 1 to n and said timestamp is any monotonically increasing sequence of numbers.

26. (Original) The program storage device of claim 18, wherein said target site is operable to apply deltas autonomously and independently from said source site.

27. (Currently Amended) The program storage device of claim ~~[[18]]~~ 19, wherein said capture and apply services send periodic signals to said monitor service to track a progression of replication for answering status and quality of service queries.

28. (Currently Amended) The program storage device of claim ~~[[18]]~~ 19, wherein said capture service selectively removes replication requests which lag other requests by more than a predetermined permissible amount.

29. (Original) The program storage device of claim 18, wherein said replicated data further comprises origination tags, wherein said origination tags are operable to prevent duplicate replications of a same data from occurring at said target site via different routes.

30. (Original) The program storage device of claim 18, wherein said apply service utilizes run-length encoding to compactly describe an interval of timestamps and sequence numbers.

31. (Currently Amended) The program storage device of claim ~~[[18]]~~ 19, ~~wherein said apply service utilizes an in-memory index when a system crash occurs and a recovery process is initiated by said distributed computing system~~ wherein communication between said capture service and said apply service occurs over an unreliable user datagram protocol (UDP) communication channel.

32. (Original) The program storage device of claim 18, wherein said target site autonomously tracks a progression of replication of said data by maintaining a separate table of applied deltas.

33. (Original) The program storage device of claim 32, wherein said separate table comprises an entry, wherein each entry in said table comprises said timestamp and said sequence number of a delta, and wherein said sequence number is operable to determine if a transaction has been misplaced in said distributed computing system.

34. (Original) The program storage device of claim 32, wherein a file-based target site can

maintain said table in a separate file and perform atomic updates by writing said file to a disk before updated files are written to said disk.

35. (Currently Amended) A data replication system comprising:

a source site ~~containing~~ comprising data to be replicated, wherein said data is embedded with replication tracking information, said replication tracking information comprising a timestamp and a contiguous sequence number;

a target site connected by a communication channel to said source site, wherein said target site is operable to receive updates using said replication tracking information, wherein said target site comprises an apply service operable to embed and analyze said tracking information during a crash recovery sequence, and wherein said apply service utilizes an in-memory index when a system crash occurs and a recovery process is initiated by said data replication system;
and

a delta production/consumption interface in communication with arbitrary data sources and targets.

36. (Original) The system of claim 35, wherein said source site comprises a capture service operable for flow control, wherein said capture service comprises a buffer.

37. (Currently Amended) The system of claim ~~[[35]]~~ 36, ~~wherein said target site comprises an apply service operable to embed and analyze said tracking information during a crash recovery sequence~~ wherein said capture service is adapted to prevent resending of an already

sent delta production/consumption value from a source site to said target site.

38. (Original) The system of claim 35, further comprising a monitor service connected to said source site and said target site operable to maintain a state of ongoing replications for status and quality-of-service tracking.

39. (Original) The system of claim 35, further comprising allowing data sources and targets of arbitrary data formats, including relational DBMSs, files, query results, XML DBMSs to be replicated, through an abstraction of delta (change) production/consumption, and a monotonically increasing timestamp on each said delta.

40. (Original) The system of claim 35, wherein said replication tracking information is used to determine if a given delta has been previously applied to said target site.

41. (Original) The system of claim 35, wherein in an event of a crash in said system, said target site requests retransmission of replicated data from said source site beginning at a given timestamp and sequence number.

42. (Original) The system of claim 35, wherein said sequence number and timestamp are operable to determine if any transaction has been lost during transmission from said source site to said target site, wherein said sequence number is a contiguous series of numbers increasing from 1 to n and said timestamp is any monotonically increasing sequence of numbers.

43. (Original) The system of claim 35, wherein said target site is operable to apply deltas autonomously and independently from said source site.

44. (Currently Amended) The system of claim [[35]] 36, wherein said capture and apply services send periodic signals to said monitor service to track a progression of replication for answering status and quality of service queries.

45. (Currently Amended) The system of claim [[35]] 36, wherein said capture service selectively removes replication requests which lag other requests by more than a predetermined permissible amount.

46. (Original) The system of claim 35, wherein said replicated data further comprises origination tags, wherein said origination tags are operable to prevent duplicate replications of a same data from occurring at said target site via different routes.

47. (Original) The system of claim 35, wherein said apply service utilizes run-length encoding to compactly describe an interval of timestamps and sequence numbers.

48. (Currently Amended) The system of claim [[35]] 36, ~~wherein said apply service utilizes an in-memory index when a system crash occurs and a recovery process is initiated by said system~~ wherein said communication channel comprises an unreliable user datagram protocol

(UDP) communication channel.

49. (Original) The system of claim 35, wherein said target site autonomously tracks a progression of replication of said data by maintaining a separate table of applied deltas.

50. (Original) The system of claim 49, wherein said separate table comprises an entry, wherein each entry in said table comprises said timestamp and said sequence number of a delta, wherein said sequence number is operable to determine if a transaction has been misplaced in said system.

51. (Original) The system of claim 49, wherein a file-based target site can maintain said table in a separate file and perform atomic updates by writing said file to a disk before updated files are written to said disk.

52. (Currently Amended) A data replication system comprising:

means for assigning a delta production/consumption value for arbitrary data sources and targets operable for replicating data;

means for embedding replication tracking information within said data; ~~and~~
means for atomically and independently applying updates at a target site using said replication tracking information[[]]; and

means for using an apply service at said target site to embed and analyze said tracking information during a crash recovery sequence, wherein said apply service utilizes an in-memory

index when a system crash occurs and a recovery process is initiated by said data replication system.